



## PEMBUATAN WEB DIREKTORI KOS UNTUK KOMUNITAS PENYEDIA JASA KOS - KOSAN MAHASISWA

Edi Satriyanto<sup>1</sup>, Wiratmoko Yuwono<sup>2</sup>, Hendra Purnomo Hadi<sup>3</sup>  
Jurusan Teknik Komputer Jaringan, Politeknik Elektronika Negeri Surabaya  
Institut Teknologi Sepuluh Nopember (ITS) Surabaya  
Kampus ITS, Keputih Sukolilo Surabaya  
Telepon +62 -31-5947280 Fax +62 -31-6946114  
Email : [hendraph@yahoo.com](mailto:hendraph@yahoo.com)

**Abstrak** – komunikasi adalah hal yang penting dan efektif di seluruh dunia dan banyak yang mempergunakannya antara lain Aplikasi Web Direktori, di dalam bidang bisnis, dalam bidang kemasyarakatan sudah banyak diimplementasikan dan terbukti memberi manfaat untuk masyarakat. Bidang bisnis pun kini sudah melihat potensi internet ini. Sekarang ini, internet menjadi sarana pembelajaran dan pertukaran informasi yang berguna untuk penyedia Jasa komunitas (provider) dan pengguna layanan kos/penyewa (consumer). Berdasarkan perkembangan teknologi dan kebutuhan masyarakat akan akses layanan publik yang praktis dan efisien, lahirlah konsep web direktori penyedia jasa sewa kos sebagai jawaban atas tuntutan tersebut.

aplikasi internet atau teknologi lain yang berkaitan di pelayanan publik yang bertujuan untuk meningkatkan akses, efisiensi, efektivitas, dan kualitas dari proses sosial dan bisnis, yang melibatkan organisasi pelayanan penyedia jasa (kos – kosan / rumah kontrak), praktisi jasa (penyedia kos), tempat, pelayanan, fasilitas, dan penyewa sebagai konsumen.

*Kata kunci - provider, consumer,web direktori.*

**Abstract** - Communication was the matter that was important and effective all over the world and many that mempergunakan him including the Web Direktori Application, in the business field, in the social field already many diimplementasikan and was proven gave the benefit for the community. The business field then currently glances at the potential for this internet. Now, the internet became means pembelajaran and the exchange of information that was useful for the community's service provider and the user of the boarding house service/the leasee (consumer). Was based on the development of technology and the requirement for the community would access of the public's service that was practical and efficient, was born the concept web the provider's directory of the coat of boarding house rent as the answer to this demand. The application of the internet or other technology that was linked in the public's service that aimed at increasing access, efficiency, the effectiveness, and the quality from the social process and the business, that involved the organisation of the service provider's service (boarded – kosan/the contract house), the service practitioner (the provider of the boarding house), the place, the service, facilities, and the leasee as the consumer.

*Kata kunci - provider, consumer,web direktori.*

### I. PENDAHULUAN

Di era yang semakin berkembang ini, teknologi informasi pun ikut berkembang dengan pesatnya. Dalam mencari data atau informasi pun menjadi lebih mudah. Karena semua data baik di pemerintahan ataupun di masyarakat mulai terkomputerisasi sehingga memudahkan dalam pencarian ataupun pemantauan data.

Saat ini internet sudah menjadi sarana komunikasi yang penting dan efektif di

seluruh dunia dan banyak bidang yang menggunakannya. Aplikasi Web Direktori dalam Segala bidang, dalam bidang bisnis, dalam bidang kemasyarakatan sudah banyak diimplementasikan dan terbukti memberi manfaat untuk masyarakat. Bidang bisnis pun kini sudah melihat potensi internet ini. Sekarang ini, internet menjadi sarana pembelajaran dan pertukaran informasi yang berguna untuk penyedia Jasa komunitas (provider) dan pengguna layanan kos/penyewa (consumer). Berdasarkan



perkembangan teknologi dan kebutuhan masyarakat akan akses layanan publik yang praktis dan efisien, lahirlah konsep web direktori penyedia jasa sewa kos sebagai jawaban atas tuntutan tersebut.

Web direktori adalah aplikasi internet atau teknologi lain yang berkaitan di pelayanan publik yang bertujuan untuk meningkatkan akses, efisiensi, efektivitas, dan kualitas dari proses sosial dan bisnis, yang melibatkan organisasi pelayanan penyedia jasa (kos – kosan / rumah kontrak), praktisi jasa (penyedia kos), tempat, pelayanan, fasilitas, dan penyewa sebagai konsumen.

Dalam proyek akhir ini dibuat sebuah web direktori yang dapat meningkatkan kemudahan untuk mengakses pelayanan jasa kos, efisiensi dan efektifitas dari segi waktu dan biaya, serta kualitas dan kuantitas dari informasi tempat tinggal, sehingga pada akhirnya dapat meningkatkan sosial masyarakat secara luas.

## II. DASAR TEORI

Diuraikan beberapa teori dasar yang mempunyai kaitan dan mendukung tentang Web Direktori Kos Untuk Komunitas Penyedia Jasa Kos - Kosan Mahasiswa

### 2.1 HTML

HyperText Markup Language (HTML) adalah sebuah bahasa markup yang digunakan untuk membuat sebuah halaman web dan menampilkan berbagai informasi di dalam sebuah browser Internet. Bermula dari sebuah bahasa yang sebelumnya banyak digunakan di dunia penerbitan dan percetakan yang disebut dengan SGML (Standard Generalized Markup Language), HTML adalah sebuah standar yang digunakan secara luas untuk menampilkan halaman web. HTML saat ini merupakan standar Internet yang didefinisikan dan dikendalikan penggunaannya oleh World Wide Web Consortium (W3C).

### 2.2 JDBC

JDBC adalah Application Programming Interface (API) yang dirancang untuk mengakses database universal berdasarkan SQL. JDBC terdiri atas JDBC 1.0 API yang memberikan fungsi-fungsi dasar untuk akses data. JDBC 2.0 API memberikan tambahan

ke fungsi-fungsi dasar dengan kelebihan-kelebihan lain yang lebih mutakhir.

JDBC adalah suatu nama trade mark, bukan sebuah singkatan. Tapi JDBC sering dikira singkatan dari Java Database Connectivity. JDBC API terdiri dari sejumlah class dan interface yang ditulis dalam bahasa Java yang menyediakan API standar sebagai alat bantu bagi pembuat program dan memberikan kemungkinan untuk menulis aplikasi database dengan menggunakan semua Java API.

JDBC API memudahkan untuk mengirim statement SQL ke sistem database relasional dan mendukung bermacam-macam bahasa SQL. Keunggulan JDBC API adalah sebuah aplikasi dapat mengakses sembarang sumber data dan dapat berjalan pada sembarang platform yang mempunyai Java Virtual Machine(JVM). Sehingga kita tidak perlu menulis satu program untuk mengakses database Sybase, Oracle atau Access dan lain-lain. Kita cukup menulis satu program yang menggunakan JDBC API, dan program dapat mengirimkan statement SQL atau statement lain ke sumber data tertentu. Dengan aplikasi yang ditulis dalam bahasa Java seseorang tidak perlu khawatir untuk menulis aplikasi yang berbeda-beda agar dapat berjalan pada platform yang berbeda-beda

### 2.3 JSP

JSP adalah suatu teknologi web berbasis bahasa pemrograman Java dan berjalan di Platform Java, serta merupakan bagian teknologi J2EE (Java 2 Enterprise Edition). JSP

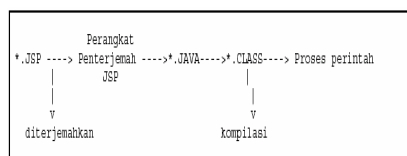
sangat sesuai dan tangguh untuk menangani presentasi di web. Sedangkan J2EE merupakan platform Java untuk pengembangan sistem aplikasi enterprise dengan dukungan API (Application Programming Interface) yang lengkap dan portabilitas serta memberikan sarana untuk membuat suatu aplikasi yang memisahkan antara business logic (sistem), presentasi dan data.

JSP merupakan bagian dari J2EE dan khususnya merupakan komponen web dari aplikasi J2EE secara keseluruhan. JSP juga memerlukan JVM (Java Virtual Machine) supaya dapat berjalan, yang berarti juga mengisyaratkan keharusan menginstal Java Virtual Machine di server, dimana JSP akan

dijalankan. Selain JVM, JSP juga memerlukan server yang disebut dengan Web Container. Teknologi JSP menyediakan cara yang lebih mudah dan cepat untuk membuat halaman-halaman web yang menampilkan isi secara dinamik. Teknologi JSP didesain untuk membuat lebih mudah dan cepat dalam membuat aplikasi berbasis web yang bekerja dengan berbagai macam web server, application server, browser dan development tool.

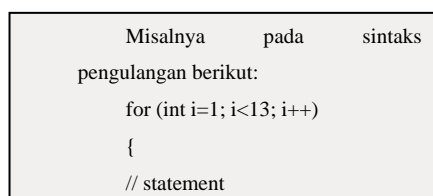
Java Server Pages (JSP) adalah bahasa scripting untuk web programming yang bersifat server side seperti halnya PHP dan ASP. JSP dapat berupa gabungan antara baris HTML dan fungsi-fungsi dari JSP itu sendiri. Berbeda dengan Servlet yang harus dikompilasi oleh USER menjadi class sebelum dijalankan, JSP tidak perlu dikompilasi oleh USER tapi SERVER yang akan melakukan tugas tersebut. Makanya pada saat user membuat pertama kali atau melakukan modifikasi halaman dan mengeksekusinya pada web browser akan memakan sedikit waktu sebelum ditampilkan.

Sebagai gambaran bagaimana JSP melalui masa hidupnya bisa dilihat pada gambar berikut :



Gambar 1 Daur hidup JSP

Seperti tipe aplikasi java lainnya (Servlet, Applet, Midlet dll), JSP juga bertipe strong Type artinya penggunaan variable pada halaman tersebut harus dideklarasikan terlebih dahulu.

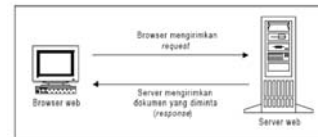


Gambar 3. contoh penggunaan variable

## 2.4 WORLD WIDE WEB (WWW)

Pada prinsipnya World Wide Web (singkatan cukup disebut "web" saja )

bekerja dengan cara menampilkan file –file HTML yang berasal dari server web pada program *client* khusus, yaitu *browser* web. Program *browser* pada *client* mengirimkan permintaan (request) kepada *server* web, yang kemudian akan dikirimkan oleh *server* dalam bentuk HTML.



Gambar 3. Konsep dasar browser dan server web

## 2.5 HYPERTEXT TRANSFER PROTOCOL (HTTP)

Server dan browser web berkomunikasi satu sama lain dengan protokol yang memang dibuat khusus untuk ini, yaitu HTTP. HTTP bertugas menangani permintaan – permintaan ( *request* ) dari *browser* untuk mengambil dokumen –dokumen web. HTTP bisa dianggap sebagai sistem yang bermodel *client-server*. *Browser* web, sebagai clientnya, mengirimkan permintaan kepada *server* web untuk menentukan apakah dokumen yang diminta bisa dikirimkan kepada *browser* atau tidak.

## 2.6 WEB CONTAINER

Menurut spesifikasi J2EE, dikenal EJB Container, Web Container dan Application Server. Web Container adalah services yang dijalankan oleh suatu Java Application Server hususnya untuk services yang compliance/kompatibel dengan Servlet dan JSP. Selain menjadi services oleh Java Application Server, Web Container dapat berdiri sendiri. Contoh Web Container adalah Tomcat, ServletExec, Resin, Jrun, Blazix. Web Container juga dapat bekerja sama dengan web server, misalnya Tomcat dengan Apache, Jrun dengan IIS.

Web Server adalah software untyk server yang menangani request melalui protokol HTTP yang digunakan oleh situs-situs web saat ini dalam menangani request file statik HTML, sepeti Apache dan Microsoft IIS. Web server sekarang sering “dibungkus” oleh Java Application Server sebagai HTTP Server. Java Application Server adalah Server yang terdiri atas HTTP Server (Web Server), EJB Container maupun Web



Container. Contoh Java Application Server: Sun J2EE RI 1.2/1.3, Borland AppServer 4.5/Enterprise Server 5.0, Oracle9i Application Server dan lainnya.

### 2.6.1 JAKARTA TOMCAT

Jakarta Tomcat adalah web application server, yang mempunyai kemampuan sebagai Servlet container dan JSP container di mana Anda bisa mendeploy Servlet dan JSP. Di atas Jakarta Tomcat, Servlet dan JSP akan bekerja melayani request dari client, yang lumrahnya adalah berupa browser.

Untuk bisa menjalankan Jakarta Tomcat, Anda membutuhkan Java Development Kit (JDK). Untuk instalasi Jakarta Tomcat, Anda bisa mendownload binary dari <http://jakarta.apache.org>, dalam format .zip, .tar.gz. Yang Anda perlu lakukan hanyalah mendecompress file tersebut.

Dalam bekerja dengan Jakarta Tomcat, Anda mempunyai sebuah directory yang dikenal sebagai TOMCAT\_HOME. TOMCAT\_HOME adalah directory di mana Jakarta Tomcat diinstall. Selanjutnya di bawah TOMCAT\_HOME Anda akan menemukan beberapa subdirectory, diantaranya bin/, conf/, logs/ dan webapp/. Di dalam subdirectory bin/ terdapat file-file executable terutama untuk menjalankan dan menghentikan Jakarta Tomcat. Di dalam subdirectory conf/ terdapat file-file untuk configuration. Di dalam subdirectory logs/ terdapat file-file log. Dan subdirectory webapp/ adalah di mana Anda bisa meletakkan aplikasi Web yang Anda bangun dengan Servlet dan JSP. Di bawah subdirectory webapp/ Anda bisa mengcreate subdirectory. Sub directory ini akan dijadikan sebagai Context oleh Jakarta Tomcat.

Anda menjalankan Jakarta Tomcat dengan mengexecute startup.sh di subdirectory bin/. Sedangkan untuk menghentikan Tomcat Anda mengexecute shutdown.sh di sub directory bin/ juga. Secara default Jakarta Tomcat siap melayani request dari client melalui port 8080. Melalui Web browser, Anda bisa menghubungi <http://localhost:8080>

### 2.6.2 JAVA VIRTUAL MACHINE

Sebelum menginstal Web Container sebagai prasyarat untuk menjalanka JSP, maka terlebih dulu harus menginstal Java Virtual

Machine. Java Virtual Machine adalah software yang berfungsi untuk menerjemahkan program Java supaya dapat dimengerti oleh komputer. Untuk memiliki Java Virtual Machine di komputer, maka perlu mendownload JDK (Java Development Kit) yang tersedia di <http://java.sun.com> karena untuk development diperlukan class-class API. Apabila tidak melakukan proses development dan hanya perlu menjalankan program, maka yang diperlukan hanya JRE (Java Runtime Environment).

### 2.6.3 MYSQL-CONNECTOR-JAVA-5.1.7

Dalam proyek akhir ini digunakan mysql-connector-java-5.1.7 sebagai media penghubung atau connection anatar data base mysql dengan JSP TomCat yang dipergunakan.

## III. SIMULASI SISTEM

### 3.1 KONEKSI DATA BASE MYSQL-JSP

Setelah mencopy connector-java-5.1.7 sebagai media connection antara mysql dengan JSP yang berada di :

```
C:\Program Files\
Java\jre1.5.0_06\lib\ext\
(File Connector di copy di folder ini)
```

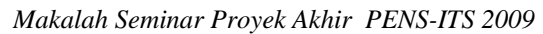
Gambar 4. copy connector-5.1.7

Setelah connector di copy dalam folder tersebut maka tomcat bisa di restart dan kemudian tag/script yang kita buat antara lain untuk membuat connection mysql - JSP

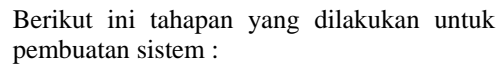
```
MySql Database Connection
Class.forName("org.gjt.mm.mysql
.Driver");
Connection con =
DriverManager.getConnection("jdbc:mysql:/
localhost:3306/dbkos","root","123456");
%>
```

Gambar 5. contoh script connection mysql

Untuk melihat hasil dari script yang kita tulis bisa di lihat di : <http://localhost:8080/kos> dimana 8080 adalah port dari Tomcat.



#### d. Pembuatan Sistem



- **Desain Antarmuka**  
Terlebih dahulu dibuat desain antar muka antara lain mengenai menu utama, dan konten menu.
- **Persiapan Data**  
Mengumpulkan data kos dan data yang digunakan adalah berupa data teks serta gambar.
- **Perancangan Konsep**  
Merencanakan bagaimana Aplikasi website kita akan diimplementasikan dalam tampilan Home kos

```

graph TD
    Guest --> Menu
    Menu --> Search
    Menu --> Exit
    Exit --> Login
    Login --> Authentication
    Authentication --> Admin[admin]
    Authentication --> Monitoring
    Admin --> Decision1{penjualan}
    Decision1 -- 0 --> Exit
    Decision1 -- 1 --> Reg_pemula[Reg. pemula]
    Reg_pemula --> Sum
    Monitoring --> Sum
    Sum --> Penjualan
    Penjualan --> Monitoring
    Monitoring --> Pemasok
    Pemasok --> Decision2{penjualan}
    Decision2 -- 0 --> Exit
    Decision2 -- 1 --> Reg_pemula
    
    subgraph Database [Data Base]
        direction TB
        Mydql
        Computer
        TomCat
        jdbc["jdbc.driver.mysql.jdbc"]
    end

```

*Gambar 8. Konsep alur web direktori kos*

setelah kita melakukan pengistallan semua software yang dibutuhkan :

1. Install *appserv-win32-2.5.8*

AppServ 2.5.8 Setup

**COMPUTER**  
php  
MySQL

## Welcome to the AppServ 2.5.8 Setup Wizard

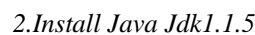
This wizard will guide you through the installation of AppServ 2.5.8.

It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.

Click Next to continue.

Next Cancel

*Gambar 9. install appserv-win32-2-2-5.8*



Version 5.0

**JAVA™ 2 Platform  
Standard Edition**

 Sun  
Microsystems

5



### 3. Install Tom Cat



Gambar 11. Tam Cat

### 4. copy connector-java-5.1.7



C:\Program Files\  
Java\jre1.5.0\_06\lib\ext\

Gambar 12. copy file connector-java-5.1.7

## BAB IV ANALISA SISTEM

Setelah perencanaan dan pembuatan Aplikasi web maka langkah selanjutnya yaitu melakukan pengujian terhadap aplikasi web ini yang telah dibuat. Pengujian dilakukan untuk mengetahui apakah aplikasi web yang dibangun dapat di aplikasikan sesuai dengan yang diinginkan.

Hasil yang dianalisa adalah report yang diminta oleh user.

Untuk menjalankan Aplikasi web direktori kos dapat di lakukan dengan mengetik pada browser <http://localhost:8080/kos> dimana 8080 adalah port dari Tomcat. Dan file yang kita buat kita simpan di folder kos.



Gambar 13. Tampilan Home kos

## BAB V PENUTUP

### 5.1 KESIMPULAN

Berdasarkan hasil analisa yang dilakukan, sistem yang telah dibangun ini masih jauh dari sempurna. Dari hasil pengujian dan pengukuran dapat disimpulkan bahwa :

- disarankan menggunakan server Tomcat berlisensi, agar tidak terjadi mudah miss connection (offline) dalam menjalankan aplikasi web ini.
- dalam penggunaan aplikasi web JSP ini akan terasa lebih berat prosesnya dibanding dengan menggunakan aplikasi yang lainnya, seperti PHP

### DARTAR PUSTAKA

1. <http://google.com>
2. <http://java.sun.com>
3. <http://www.kostjogja.com>
4. <http://ilmukomputer.com>
5. <http://planet-source-code.com>
6. <http://reseindia.net>